On Channel Failures, File Fragmentation Policies, and Heavy-Tailed Completion Times

Jayakrishnan Nair, Martin Andreasson, Lachlan L. H. Andrew, Steven H. Low and John C. Doyle

***Abstract*—It has been recently discovered that heavy-tailed completion times can result from protocol interaction even when file sizes are light-tailed. A key to this phenomenon is the use of a restart policy where if the file is interrupted before it is completed, it needs to restart from the beginning. In this paper, we show that fragmenting a file into pieces whose sizes are either bounded or independently chosen after each interruption guarantees light-tailed completion time as long as the file size is light-tailed; i.e., in this case, heavy-tailed completion time can only originate from heavy-tailed file sizes. If the file size is heavy-tailed, then the completion time is necessarily heavy-tailed. For this case, we show that when the file size distribution is regularly varying, then under independent or bounded fragmentation, the completion time tail distribution function is asymptotically bounded above by that of the original file size stretched by a constant factor. We then prove that if the distribution of times between interruptions has non-decreasing failure rate, the expected completion time is minimized by dividing the file into equal sized fragments; this optimal fragment size is unique but depends on the file size. We also present a simple blind fragmentation policy where the fragment sizes are constant and independent of the file size and prove that it is asymptotically optimal. Both these policies are also shown to have desirable completion time tail behavior. Finally, we bound the error in expected completion time due to error in modeling of the failure process.**

I. Motivation and summary

It has been recently discovered that heavy-tailed file completion time can result from protocol interaction even when the file size is light-tailed, provided its distribution has infinite support [3]–[6]. Indeed, the completion time can be heavy-tailed even when the file size has a tail that decays exponentially or superexponentially. A key to this phenomenon is the RESTART feature [3], [4] where if a file is interrupted in the middle of its processing, the entire file needs to restart from the beginning, i.e., the work that is partially completed is lost.

A standard mechanism for reducing completion times in an unreliable service environment is fragmentation. For example, a file to be transmitted over an unreliable channel is fragmented into packets. Similarly, in computing environments, a file/job is fragmented by introducing checkpoints [7]–[9]. The purpose of such fragmentation is of course that when a server failure occurs, the only work lost corresponds to the fragment being currently processed.

In this paper, we are motivated by the question: Can fragmentation 'lighten' the completion time tail? The main contribution of this paper is show that the completion time tail is indeed 'lightened' by a large class of fragmentation policies.

In particular, we consider a model for file transfer over an unreliable channel and propose fragmentation policies that guarantee light-tailed completion time for light-tailed file sizes. In the models of [3]–[6], heavy-tailed completion time seems to arise from repeated comparison of a sequence of independent, identically distributed (i.i.d.) random variables (server/channel availability periods) with the *same* random variable (original job/file size) that has an *infinite support*. This motivates fragmentation policies that avoid this character. Specifically, we consider policies that partition files into fragments with independent, or bounded sizes; note that packet sizes are naturally bounded by network hardware. We show that these policies produce a light-tailed completion time as long as the original file size is light-tailed, i.e., in this case, a heavy-tailed file completion time can only originate from a heavy-tailed file size (Section III). If the file size is heavy-tailed, then the file completion time is necessarily heavy-tailed. In this case, we show that if the file size distribution is regularly varying, then under independent or bounded fragmentation, the completion time tail distribution function is asymptotically bounded above by that of the original file size stretched by a constant factor. This means that in the degree sense, the completion time distribution is only as heavy-tailed as the file size distribution.

While the above results pertain to the *tail* of the completion time, another natural (and complementary) metric to consider is the *expected* completion time. We prove that if the failure distribution has a non-decreasing failure rate, it is optimal (for the expected completion time) to divide the file into equal sized fragments, whose size depends on the file size (Section IV-A). We also present a simple blind fragmentation policy where the fragment size is constant and independent of the file size and prove that its expected file completion time is asymptotically optimal (Section IV-B). The optimal policy as well as the suboptimal blind policy create bounded fragments, and therefore also produce desirable completion time tail behavior. Next, we present simple bounds on the error in expected completion time when there is error in modeling the failure process (Section V).

Finally, we study a related model for job checkpointing in a computing environment, and show that our main results for

Jayakrishnan Nair is with the Department of Electrical Engineering, IIT Bombay, India

Martin Andreasson is with the ACCESS Linnaeus Centre, KTH Royal Institute of Technology, Sweden

Lachlan Andrew is with the Centre for Advanced Internet Architectures, Swinburne University of Technology, Australia

Steven Low and John Doyle are with the Computing and Mathematical Sciences Department, California Institute of Technology, USA.

A shorter version of this paper appeared in IEEE INFOCOM [1]. A preliminary version was also presented at the MAMA workshop [2].

the file fragmentation model can be extended to this setting (Section VI).

II. Model and preliminaries

A. Model

Consider a file with a possibly random size $L > 0$. The file is fragmented into packets which are then sent over an unreliable channel with unit transmission rate. A packet contains a fragment of the file and a fixed-sized overhead (header, trailer). The larger the packet size, the more likely the transmission is to fail. This will be the case, e.g., if the channel randomly introduces independent bit errors so a packet with more bits has a higher probability of being corrupted and needing a retransmission; see [10, p. 132] for such a failure model for satellite and terrestrial communications. More generally, for the nth transmission attempt, let $x_n + \phi$ be the packet size, where x_n is the size of the file fragment and ϕ is the constant overhead. All sizes are measured in terms of the transmission time over the channel with unit rate. Let $(A_n, n = 1, 2, \dots)$ be i.i.d. non-negative random variables with common distribution F and independent of L, with $P(A_1 > \phi) > 0$. The nth transmission attempt will be successful if and only if $A_n \geq x_n + \phi$.[1]

To formulate the problem precisely, we abuse notation and use $x = (x_n, n = 1, 2, \dots)$ to denote both the control (fragmentation) policy and the fragment sizes under the policy, depending on the context. Let the state $l_n := l_n^x$ be the remaining file size just after the start of the nth transmission under control policy x. Then the state l_n evolves according to,

$$l_{n+1} = l_n - x_n \, \mathbf{1}(A_n \geq x_n + \phi), \quad n = 1, 2, \dots \quad (1)$$

$$l_1 = L \quad (2)$$

where $\mathbf{1}(z) = 1$ if z is true and 0 otherwise. We implicitly restrict ourselves to admissible policies x under which $0 \leq x_n \leq l_n$ for all n. We emphasize that the state sequence $(l_n, n \geq 1)$ depends on the control policy $x = (x_n, n \geq 1)$ though this is not explicit in the notation. The time between the nth and the $n+1$st submission is the cost at the nth stage and is given by:

$$\tau_n := (x_n + \phi) \, \mathbf{1}(l_n > 0) \quad (3)$$

Clearly, the transmission time sequence $(\tau_n, n \geq 1)$ also depends on the control x. Let $T(L)$ be the file completion time under control x as a function of the initial file size L;

$$T(L) := T^x(L) := \sum_{n \geq 1} \tau_n. \quad (4)$$

In summary, our file fragmentation model is specified by (1)–(4) with the i.i.d. random sequence $(A_n, n \geq 1)$. In subsequent sections, we will study the impact of the choice of the fragment sizes $(x_n, n = 1, 2, \dots)$ on the file completion time.

[1]We note that A_n does not need to be interpreted as a channel availability period. Essentially, our model assumes that each packet transmission independently succeeds with a probability that is a non-increasing function of the packet size. The random variable A_n simply captures the randomness of the channel that affects the nth packet transmission.

Our model is an adaptation of the model in [3]–[6] where a server alternates between availability periods and unavailability periods. There, the server availability periods have durations $(A_n, n \geq 1)$ that are i.i.d. random variables. The unavailability periods have durations $(U_n, n \geq 1)$ that are i.i.d. and independent of $(A_n, n \geq 1)$. Without fragmentation, the entire file is submitted at the beginning of each availability period until it completes successfully, $x_n = L$ for all n. Our model here has $U_n = 0$; furthermore, the one-stage cost for an unsuccessful fragment submission is $x_n + \phi$ in our case but $A_n + U_n$ in theirs. Thus, our model captures the scenario in which the sender is informed of the failure only after the entire packet has been sent. In contrast, in the model of [3]–[6], the sender is immediately informed of a server failure (note that in this model, A_n has an interpretation as a server availability period). However, these differences do not qualitatively change our conclusions; indeed, we present a parallel set of results in Section VI for a job checkpointing model that is closer to the model in [3]–[6].

B. Notation and preliminaries

Throughout this paper, $\overline{\lim}$ denotes the limit superior, $\underline{\lim}$ the limit inferior and $\mathbb{E}[\cdot]$ the expectation. For any functions $\gamma(t)$ and $\lambda(t)$,

1) $\gamma(t) \sim \lambda(t)$ means $\lim_{t\to\infty} \gamma(t)/\lambda(t) = 1$,
2) $\gamma(t) \lesssim \lambda(t)$ means $\overline{\lim}_{t\to\infty} \gamma(t)/\lambda(t) \leq 1$,
3) $\gamma(t) = o(\lambda(t))$ means $\lim_{t\to\infty} \gamma(t)/\lambda(t) = 0$.

Consider non-negative random variables X and Y. Let $X \stackrel{d}{=} Y$ denote that X and Y are equal in distribution. We will use the notation $X \leq_{\text{a.s.}} Y$ to mean $X \leq Y$ almost surely. The notation $X \leq_{\text{st}} Y$ means X is stochastically dominated by Y, i.e., $P(X > t) \leq P(Y > t)$ for all $t \geq 0$. It is easy to see that $X \leq_{\text{a.s.}} Y$ implies $X \leq_{\text{st}} Y$. The following lemma will be useful later.

Lemma 1. *If random variables A, B, C satisfy $A \leq_{\text{st}} B \leq_{\text{st}} C$, and $P(A > x) \sim P(C > x)$, then*

$$P(A > x) \sim P(B > x) \sim P(C > x).$$

The elementary proof is omitted. Let $G(x) = P(X \leq x)$ denote the distribution function (df) of the non-negative random variable X and $\overline{G}(x) := 1 - G(x)$ denote its tail df.

Definition 1. *The df G (or the random variable X) is said to be* heavy-tailed *if $\overline{\lim}_{x\to\infty} e^{\theta x}\overline{G}(x) = \infty$ for all $\theta > 0$. The df G (or the random variable X) is said to be* light-tailed *if it is not heavy-tailed, i.e., if there exists a $\theta > 0$ such that $\lim_{x\to\infty} e^{\theta x}\overline{G}(x) = 0$.*

Intuitively, a distribution is heavy-tailed if its tail df is (asymptotically) heavier than that of any exponential distribution. Conversely, a distribution is light-tailed if its tail df is (asymptotically) dominated by that of some exponential distribution. The following lemma describes some closure properties of the class of light-tailed distributions we will use in this paper.

Lemma 2. *[Closure properties of light-tailed distributions]*

1) *Let X, Y be non-negative random variables satisfying $X \leq_{\rm st} Y$. If Y is light-tailed, then X is light-tailed.*
2) *Let X, Y be non-negative random variables. If X, Y are light-tailed, then $X + Y$ is light-tailed.*
3) *Let $(X_i, i \geq 1)$ be a sequence of non-negative i.i.d. light-tailed random variables, and N be an integer random variable. If N is light-tailed, then the random sum $\sum_{i=1}^{N} X_i$ is light-tailed.*
4) *Let L be a non-negative random variable and $\{X_i\}_{i\geq 1}$ a sequence of non-negative i.i.d. random variables independent of L and satisfying $P(X_i > 0) > 0$. If L is light-tailed, so is $\inf\{n | \sum_{i=1}^{n} X_i \geq L\}$.*

We give the proof of this lemma in Appendix A.

An important class of heavy-tailed distributions is the class of regularly varying distributions (see [11], Chapter 2 of [12]).

Definition 2. *A df G is regularly varying with index/degree $\alpha > 0$ (denoted $G \in \mathcal{RV}(\alpha)$) if*

$$\overline{G}(x) = x^{-\alpha}\chi(x)$$

where $\chi(x)$ is a slowly varying function, i.e., $\chi(x)$ satisfies

$$\lim_{x\to\infty} \frac{\chi(xy)}{\chi(x)} = 1 \quad \forall y > 0.$$

We will abuse notation and use $L \in \mathcal{RV}(\alpha)$ to mean the df G_L of a random variable L is in $\mathcal{RV}(\alpha)$. Regularly varying distributions are a generalization of the class of Pareto distributions, also referred to as power-law distributions or Zipf distributions. Note that a smaller value of α implies a heavier tail. The following lemmas pertaining to regular variation will be useful in our proofs.

Lemma 3. *Consider non-negative random variables X, Y. If $X \in \mathcal{RV}(\alpha)$ and $P(X > t) \sim P(Y > t)$, then $Y \in \mathcal{RV}(\alpha)$.*

The proof follows from the definition.

Lemma 4. *If $X \in \mathcal{RV}(\alpha)$, then $P(X > t) \sim P(X > t + c)$ for all $c \in \mathbb{R}$.*

This lemma is a consequence of the fact that regularly varying distributions are a sub-class of the class of long-tailed distributions; see [13].

Lemma 5. *If $\chi(x)$ is slowly varying, then*

$$\lim_{x\to\infty} x^{\beta}\chi(x) = \begin{cases} \infty & \text{if } \beta > 0 \\ 0 & \text{if } \beta < 0 \end{cases}.$$

See Prop. 2.6 in [12] for a proof. Lemma 5 leads to the following corollary.

Corollary 1. *If X for $\mathcal{RV}(\alpha)$, then for any $\epsilon > 0$,*

$$t^{-(\alpha+\epsilon)} \leq P(X > t) \leq t^{-(\alpha-\epsilon)}$$

for large enough t.

III. Completion time tail asymptotics

In this section, we study the tail behavior of the completion time under a broad class of fragmentation policies. To motivate our results, we first state the following lemma, which considers the case of no fragmentation.

Lemma 6 ([3]–[6])**.** *Without fragmentation (i.e., $x_n = L$ until the whole file is transmitted successfully), $T(L)$ is heavy-tailed as long as L has infinite support.*

The proof follows from Lemma 1 in [6]. Lemma 6 implies that without fragmentation, the completion time $T(L)$ can be heavy-tailed even for light-tailed file sizes, e.g., file size distributions with an exponential or even superexponential tail df. Intuitively, this is because large files need to be retransmitted many times, and therefore have a disproportionately large completion time. Our results in this section (Theorems 1–3) imply that under a broad class of fragmentation policies, the completion time $T(L)$ is light-tailed provided L is light-tailed. Thus, with these policies, *heavy-tailed completion times can only arise from heavy-tailed file sizes.*

Of course, if the file size distribution is heavy-tailed, then the completion time is necessarily heavy-tailed. For a regularly varying file size distribution, the following lemma tells us *how* heavy the completion time tail is with no fragmentation, under the additional assumption that A_1 is light tailed.

Lemma 7. *Suppose $L \in \mathcal{RV}(\alpha)$, and A_1 is light-tailed. Without fragmentation (i.e., $x_n = L$ until the whole file is transmitted successfully),*

$$\lim_{t\to\infty} \frac{-\log P(T(L) > t)}{\log(t)} = 0.$$

The proof of this lemma follows easily from the arguments in the proof of Theorem 2 in [6]. Lemma 7 implies that for any $\epsilon > 0$, $P(T(L) > t) \geq t^{-\epsilon}$ for large enough t, which means that $P(L > t) = o(P(T(L) > t))$ (see Corollary 1). Thus, the completion time tail is asymptotically heavier than the file size tail. In contrast, the results in this section (Theorems 1–3) imply that under the above mentioned broad class of fragmentation policies, the tail df of $T(L)$ is bounded above by a scaled version of the tail df of L. This means that in the degree sense, the completion time is only as heavy-tailed as the file size.

A. Results

We now define the three classes of fragmentation policies studied in this section.

- **Independent fragmentation:** $x_n = \min\{X_n, l_n\}, n \geq 1$, where $(X_n, n \geq 1)$ is a sequence of i.i.d. strictly positive light-tailed random variables independent of L and $(A_n, n \geq 1)$ such that $P(A_1 \geq X_1 + \phi) > 0$. Note that x_n is the size of the fragment in the nth transmission attempt. If a fragment is not successfully transmitted, then that specific fragment is not retransmitted; instead a new fragment is selected starting from the same point in the file.
- **Bounded fragmentation:** x_n satisfies $\min\{b, l_n\} \leq x_n \leq \min\{c, l_n\}$, $n \geq 1$, for some constants $0 < b \leq c$ such that $P(A_1 \geq c + \phi) > 0$. Note that the choice of x_n may be random and may depend on L.
- **Constant fragmentation:** $x_n = \min\{b, l_n\}$ for some deterministic constant $b > 0$ satisfying $P(A_1 \geq b + \phi) > 0$. This is a special case of independent fragmentation and of bounded fragmentation.

We now state our results for each of these classes.

Theorem 1 (Independent fragmentation)**.** *Under an independent fragmentation policy*

1) *If L is light-tailed, then $T(L)$ is light-tailed.*
2) *If $L \in \mathcal{RV}(\alpha)$, then*

$$P(L > t) \leq P(T(L) > t) \lesssim P\left(L > \frac{t}{\sigma}\right)$$

where

$$\sigma = \frac{\mathbb{E}[X_1] + \phi}{P(X_1 + \phi \leq A_1)\,\mathbb{E}[X_1 | X_1 + \phi \leq A_1]}. \quad (5)$$

The next result says that any policy that does not choose arbitrarily large or arbitrarily small fragment sizes produces light-tailed completion time provided L is light-tailed.

Theorem 2 (Bounded fragmentation)**.** *Under a bounded fragmentation policy*

1) *If L is light-tailed, then $T(L)$ is light-tailed.*
2) *If $L \in \mathcal{RV}(\alpha)$, then*

$$P(L > t) \leq P(T(L) > t) \lesssim P\left(L > \frac{t}{\sigma}\right)$$

where

$$\sigma = \frac{c + \phi}{bP(A_1 \geq c + \phi)}.$$

Intuitively, if packet size is too small, the overhead can dominate the transmission, reducing efficiency. If the packet is too large, the failure probability can be too high. Hence we consider a policy "reasonable" if the fragments it selects are neither too small nor too large. Theorem 2 then guarantees that any reasonable fragmentation policy 'lightens' the completion time tail, relative to the case of no fragmentation.

Since constant fragmentation is a special case of independent and bounded fragmentation, Theorems 1 and 2 imply that under constant fragmentation, $T(L)$ is light-tailed if L is light-tailed. When L is regularly varying, constant fragmentation provides the following sharper characterization of the asymptotics: $T(L)$ is regularly varying with the same degree.

Theorem 3 (Constant fragmentation)**.** *Under a constant fragmentation policy*

1) *If L is light-tailed, then $T(L)$ is light-tailed.*
2) *If $L \in \mathcal{RV}(\alpha)$, then $P(T(L) > t) \sim P\left(L > \frac{t}{g(b)}\right)$ where*

$$g(x) = \frac{x + \phi}{xP(A_1 \geq x + \phi)}.$$

Theorem 3 motivates choosing the constant fragment size $a := \arg\min_{x>0} g(x)$. Within the class of constant fragmentation policies, this choice produces in some sense the lightest possible completion time tail asymptotics. We will prove in Section IV-B that this policy also almost minimizes the expected completion time.

B. Proofs of Theorems 1–3

The proofs of Theorems 1–3 rely on the following.

Lemma 8. *Let L be a random variable, and $(X_n, n \geq 1)$ be a sequence of i.i.d. strictly positive light-tailed random variables independent of L and $(A_n, n \geq 1)$ such that $P(A_1 > X_1 + \phi) > 0$. Let*

$$\begin{aligned} Y_n &:= X_n \mathbf{1}(X_n + \phi \leq A_n), \\ M &:= \inf\left\{m : \sum_{n=1}^{m} Y_n \geq L\right\}, \quad (6) \\ \tilde{T}(L) &:= \sum_{n=1}^{M} (X_n + \phi). \quad (7) \end{aligned}$$

1) *If L is light-tailed, then $\tilde{T}(L)$ is light-tailed.*
2) *If $L \in \mathcal{RV}(\alpha)$, then $P\left(\tilde{T}(L) > t\right) \sim P(L > t/\sigma)$ where σ is given by (5).*

The proof of this lemma for the case of regularly varying L is based on the following lemma, proved in [14].

Lemma 9 ([14])**.** *Let $L \in \mathcal{RV}(\alpha)$. For $t \geq 0$, let $R(t)$ be a non-negative, almost surely non-decreasing stochastic process independent of L satisfying the following conditions:*

1) *For some $\gamma \in (0, 1)$, $\lim_{t\to\infty} R(t)/t = \gamma$ a.s..*
2) *For some positive finite constant K, $P(R(t)/t < K) = o(P(L > t))$.*

Then $P(L > R(t)) \sim P(L > \gamma t)$.

Proof of Lemma 8: We consider the cases of light-tailed and regularly varying L separately.

Case 1: L is light-tailed. Under the assumptions of the lemma, $(Y_n, n \geq 1)$ is an i.i.d. sequence satisfying $P(Y_1 > 0) > 0$. Invoking Lemma 2(4), we conclude from (6) that M is light-tailed. It follows that $\tilde{T}(L)$ is light-tailed from (7) invoking Lemma 2(3).

Case 2: $L \in \mathcal{RV}(\alpha)$. Let $N(t) := \sup\{n : \sum_{i=1}^{n}(X_i + \phi) \leq t\}$, $R(t) := \sum_{i=1}^{N(t)} Y_i$. Note that $P\left(\tilde{T}(L) > t\right) = P(R(t) < L)$. To complete the proof, it suffices to show that the process $R(t)$ satisfies conditions (1) and (2) of Lemma 9 with $\gamma = 1/\sigma$.

Condition (1) of Lemma 9 is verified using the renewal reward theorem.

$$\lim_{t\to\infty} \frac{R(t)}{t} = \frac{\mathbb{E}[Y_1]}{\mathbb{E}[X_1 + \phi]} = \frac{1}{\sigma}$$

almost surely. Note that $\sigma > 1$ since $\phi > 0$. To verify Condition (2), pick $K \in (0, 1/\sigma)$. Since $K < 1/\sigma$, we can find $\eta, \nu > 0$ such that $K = \eta\nu$, $\eta < \mathbb{E}[Y_1]$ and $\nu < 1/\mathbb{E}[X_1 + \phi]$.

Then

$$\begin{aligned}
P(R(t) < Kt) &= P\left(\sum_{i=1}^{N(t)} Y_i < Kt\right) \\
&= P\left(N(t) < t\nu\right) - P\left(\sum_{i=1}^{N(t)} Y_i \geq Kt \wedge N(t) < t\nu\right) \\
&\quad + P\left(\sum_{i=1}^{N(t)} Y_i < Kt \wedge N(t) \geq t\nu\right) \\
&\leq P\left(N(t) < t\nu\right) + P\left(\sum_{i=1}^{N(t)} Y_i < Kt \wedge N(t) \geq t\nu\right) \\
&\leq P\left(\sum_{i=1}^{\lfloor t\nu \rfloor}(X_i + \phi) \geq t\right) + P\left(\sum_{i=1}^{\lceil t\nu \rceil} Y_i < Kt\right) \\
&\leq P\left(\sum_{i=1}^{\lfloor t\nu \rfloor}(X_i + \phi) \geq \frac{\lfloor t\nu \rfloor}{\nu}\right) + P\left(\sum_{i=1}^{\lceil t\nu \rceil} Y_i < \eta \lceil t\nu \rceil\right).
\end{aligned}$$

Noting that $1/\nu > \mathbb{E}\left[X_1 + \phi\right]$ and $\eta < \mathbb{E}\left[Y_1\right]$, and that X_1, Y_1 are light-tailed, we can use the Chernoff bound to argue that there exist positive constants C, λ such that for large enough t,

$$P\left(R(t) < Kt\right) \leq Ce^{-\lambda t}.$$

Since $P(L > t) = t^{-\alpha}\chi(t)$ for slowly varying χ, this implies

$$\overline{\lim_{t\to\infty}} \frac{P\left(R(t) < Kt\right)}{P\left(L > t\right)} \leq \overline{\lim_{t\to\infty}} \frac{Ce^{-\lambda t}}{t^{-\alpha}\chi(t)} = \overline{\lim_{t\to\infty}} \frac{Ct^{\alpha+1}e^{-\lambda t}}{t\chi(t)} = 0.$$

The last step above uses Lemma 5. It follows that $P\left(R(t) < Kt\right) = o(P\left(L > t\right))$. This completes the proof. ∎

We are now ready to prove Theorems 1–3.

Proof of Theorem 1: Consider the completion time $\tilde{T}(L)$ under the policy $\tilde{x}_n := X_n$. Clearly $T(L) \leq_{\text{a.s.}} \tilde{T}(L)$.

If L is light-tailed, then from Lemma 8, we conclude that $\tilde{T}(L)$ is light-tailed, which implies $T(L)$ is light-tailed (Lemma 2(1)).

If $L \in \mathcal{RV}(\alpha)$, then from Lemma 8, we conclude that $P\left(\tilde{T}(L) > t\right) \sim P\left(L > \frac{t}{\sigma}\right)$. Since $T(L) \leq_{\text{a.s.}} \tilde{T}(L)$, it follows that $P\left(T(L) > t\right) \lesssim P\left(L > \frac{t}{\sigma}\right)$. ∎

Proof of Theorem 2: Define $\tilde{L} := cL/b$. With file size $\tilde{L}$, consider the policy $\tilde{x}_n = \min\{c, \tilde{l}_n\}$, $n \geq 1$, where $\tilde{l}_1 = \tilde{L}$, $\tilde{l}_n$ denotes the remaining file size just after the nth submission. Note that this policy satisfies the conditions of Theorem 1 with $x_n = c$. Denote the completion time under this scheme by $T^c(\tilde{L})$.

We will now argue that $T(L) \leq_{\text{a.s.}} T^c(\tilde{L})$. Consider a sample path, determined by the realization of L, $(A_n, n \geq 1)$ and the fragment sizes $(x_n, n \geq 1)$. For any n, if fragment submission $\tilde{x}_n$ succeeds, then submission x_n succeeds. Hence $l_n \leq b\tilde{l}_n/c$ for all $n \geq 1$. This implies $T(L) \leq T^c(\tilde{L})$.

If L is light-tailed, so is $\tilde{L}$. Theorem 1 then implies that $T^c(\tilde{L})$ is light-tailed, which implies $T(L)$ is light-tailed (Lemma 2(1)).

If $L \in \mathcal{RV}(\alpha)$, then $\tilde{L} \in \mathcal{RV}(\alpha)$. Theorem 1 implies that

$$\begin{aligned}
P\left(T^c(\tilde{L}) > t\right) &\lesssim P\left(\tilde{L} > \frac{tcP\left(A_1 \geq c + \phi\right)}{c + \phi}\right) \\
&= P\left(L > \frac{tbP\left(A_1 \geq c + \phi\right)}{c + \phi}\right) \\
&= P\left(L > \frac{t}{\sigma}\right).
\end{aligned}$$

Since $T(L) \leq_{\text{a.s.}} T^c(\tilde{L})$, we have $P\left(T(L) > t\right) \lesssim P\left(L > \frac{t}{\sigma}\right)$. ∎

Proof of Theorem 3:

Since constant fragmentation is a special case of independent and bounded fragmentation, the proof for the case of light-tailed L follows directly from Theorems 1 or 2.

Assume then that $L \in \mathcal{RV}(\alpha)$. We will invoke Lemma 8 with $X_n := b$, $n \geq 1$. Define

$$\hat{L} := b\left\lfloor \frac{L}{b} \right\rfloor, \quad \tilde{L} := b\left\lceil \frac{L}{b} \right\rceil.$$

It is easy to see that

$$\tilde{T}(\hat{L}) \leq_{\text{a.s.}} T(L) \leq_{\text{a.s.}} \tilde{T}(\tilde{L}).$$

We will now argue that $\hat{L}, \tilde{L} \in \mathcal{RV}(\alpha)$. Clearly,

$$\max\{L - b, 0\} \leq_{\text{a.s.}} \hat{L} \leq_{\text{a.s.}} L \leq_{\text{a.s.}} \tilde{L} \leq_{\text{a.s.}} L + b.$$

Using Lemma 4, we see that $P\left(\max\{L - b, 0\} > t\right) \sim P\left(L + b > t\right)$. This implies, using Lemma 1, that

$$P\left(\hat{L} > t\right) \sim P\left(L > t\right) \sim P\left(\tilde{L} > t\right),$$

which in turn implies $\hat{L}, \tilde{L} \in \mathcal{RV}(\alpha)$ (see Lemma 3). By Lemma 8, we see that

$$P\left(\tilde{T}(\hat{L}) > t\right) \sim P\left(\tilde{T}(\tilde{L}) > t\right) \sim P\left(L > \frac{t}{g(b)}\right).$$

This implies $P\left(T(L) > t\right) \sim P\left(L > \frac{t}{g(b)}\right)$ by Lemma 1. ∎

IV. Minimizing the average completion time

studied the tail asymptotics of the completion time; in this section, we turn our attention to its mean. Specifically, under the assumption that F has a non-decreasing failure rate, we derive the fragmentation policy that minimizes the expected completion time. We show that this policy divides the file into equal sized fragments, whose size depends on the file size L, but for all L remains close to a value determined solely by F. We also present a fragmentation policy that is blind to the file size, but is asymptotically optimal. We show that under both these policies, the completion time is light-tailed so long as L is light-tailed. If L is regularly varying, then the completion time is regularly varying with the same index.

Consider

$$\min_x \ \mathbb{E}\left[T^x(L)\right] := \min_x \ \left(\lim_{N\to\infty} \ \mathbb{E}\left[\sum_{n=1}^{N} \tau_n \mid l_1 = L\right]\right) \quad (8)$$

An *optimal policy* is one that achieves the minimum of (8). We will restrict ourselves to the class of stationary Markov policies where the decision at time n depends only on the state l_n and

not on the time n nor on past states. Since any optimal policy will never choose fragment sizes x_n with $P(A_1 \geq x_n + \phi) = 0$, we will assume without loss of generality that $P(A_1 \geq x_n + \phi) > 0$ for the class of policies that we consider. Our discussion in this section (except in IV-C, which deals with completion time tail asymptotics) will be for some generic realization of the initial file size $L > 0$.

A. Optimal policy

A stationary Markov policy is a function $x(l)$ of the remaining file size l with the following interpretation. Given l, a packet of size $x(l) + \phi$ is formed. If the packet is successfully transmitted, the remaining file size will be $l - x(l)$. If the transmission fails, the file size remains unchanged and therefore the next fragment remains $x(l)$, until the packet is successfully transmitted. Recall that F is the distribution function of A_i. The expected time it takes to successfully transmit a fragment is $(x(l)+\phi)/\overline{F}(x(l)+\phi)$, the cost per trial multiplied by the expectation of the number of trials, which is geometrically distributed with parameter $F(x(l) + \phi)$. This implies that if we let $J(l) := \mathbb{E}[T(l)]$ denote the expected completion time when the file size is l under a generic Markov policy $x(l)$, then

$$J(l) \quad = \quad J(l - x(l)) + \frac{x(l) + \phi}{\overline{F}(x(l) + \phi)}.$$

Given any Markov policy $x(l)$, consider the sequence of fragments $x_1, x_2, \ldots$, generated from an initial file size L, defined recursively as:

$$x_1 := x(L);\ x_{i+1} := x(L - x_i), i \geq 1$$

such that $\sum_k x_k = L$. Define the expected time to successfully transmit a segment of size x as

$$h(x) \quad = \quad \frac{x + \phi}{\overline{F}(x + \phi)}. \tag{9}$$

The expected completion time is thus

$$J(L) \quad = \quad \sum_k h(x_k).$$

Since $h(x) \geq h(0) > \phi > 0$ for all $x \geq 0$, an optimal policy must only have finitely many terms in $J(L)$. Let $J^*(L)$ denote the (minimum) expected completion time under an optimal policy x^*.

Consider the following optimization problem:

$$H^* := \min_K \min_{y_1,\ldots,y_K} \quad \sum_{k=1}^K h(y_k) \tag{10a}$$

$$\text{subject to} \quad \sum_{k=1}^K y_k = L \tag{10b}$$

$$y_k > 0, \quad k = 1, \ldots, K \tag{10c}$$

$$K = 1, 2, \ldots. \tag{10d}$$

We now argue that, given $L > 0$, the sequence of fragment sizes $x^* := (x_1^*, x_2^*, \ldots, x_{K^*}^*)$ generated by a Markov policy $x^*(l)$ minimizes the expected completion time $\mathbb{E}[T(L)]$ if and only if (K^*, x^*) is a minimizer of (10a)–(10d). Now, any finite K and sequence $(x_1, x_2, \ldots, x_K)$ with $\sum_{k=1}^K x_k = L$, $x_k > 0$ is a feasible solution of (10a)–(10d). Hence, $H^* \leq J^*(L)$. Conversely, given any minimizer (K^*, y^*) of (10a)–(10d), we will exhibit a Markov policy $x(l)$ that generates the sequence of fragment sizes that coincide with the given $y^* = (y_1^*, \ldots, y_{K^*}^*)$. This implies the minimum expected completion time satisfies $J^*(L) \leq H^*$. Hence, $J^*(L) = H^*$.

We can thus focus on solving (10a)–(10d). Indeed, we will show that under Assumption A1 below, (10a)–(10d) has a unique solution with $x_n^* = x^*$ for all n, implying that the optimal policy divides the file into equal sized fragments.[2]

Parametrize the optimization problem (10a)–(10d) by the file size in (10b), and write any minimizer as $(K^*(l), y^*(l))$ when the file size is l. Consider the Markov policy $x(l)$ that solves (10a)–(10d) with file size l and selects the segment size $x(l) = y_1^*(l)$, i.e., the policy uses the first element of the solution $y^*(l)$ as the segment size when the remaining file size is l. The next segment size under policy $x(l)$ therefore comes from the solution of (10a)–(10d) with file size $l - x(l)$, i.e., $x(l - x(l)) = y_1^*(l - y_1^*(l))$. But $y_1^*(l - y_1^*(l))$ must be (equal to) the second element in the original solution, i.e., $y_1^*(l - y_1^*(l)) = y_2^*(l)$, for otherwise, $y^*(l)$ could not have been a minimizer. This implies by induction that the Markov policy $x(l)$ generates the sequence of fragment sizes from L that coincides with (K^*, y^*).

The main result of this section is the following theorem that says that the optimal policy creates equal sized fragments. The optimal fragment size depends on the file size. Define

$$g(x) = \frac{x + \phi}{x\overline{F}(x + \phi)} \tag{11}$$

and

$$a = \arg\min_x g(x), \qquad x \in \mathbb{R}_+ \tag{12}$$

Note that $g(x) = h(x)/x$ where $h(x)$ is the expected cost (time) to successfully transmit a segment of size x defined in (9). Hence we can interpret $g(x)$ as the per-bit cost for a fragment of size x, and a as the fragment size that minimizes the per-bit cost. It will become clear below that the optimal fragment size x^* is close to a and the minimum cost $J^*(L)$ is close to $Lg(a)$, under the following assumption:

Assumption A1: The density function $F' =: f$ exists. Moreover, the failure rate $\lambda(x) := f(x)/\overline{F}(x)$ is continuous and non-decreasing.[3]

Theorem 4 (Optimal fragmentation)**.** *Under Assumption A1, for any $L > 0$, minimizers (K^*, x^*) of (10) are given by:*

1) *K^* equals $\lfloor L/a \rfloor$ or $\lceil L/a \rceil$ whichever produces a smaller value of $g(L/K^*)$.*
2) *$x_k^* = L/K^*$ for $k = 1, \ldots, K^*$.*

Therefore, the optimal policy divides the file into K^* fragments of equal size. Each fragment is (re)submitted to the channel until the transmission is successful.

[2]We abuse notation and use x to denote a fragmentation policy, a vector of fragment sizes, or a scalar representing a constant fragment size, depending on the context; x^* denotes these quantities under an optimal policy.

[3]If $f(x) = \overline{F}(x) = 0$, define $\lambda(x) = \infty$.

Proof of Theorem 4: We will first prove that, given any K, the minimizer x^* of the inner minimization exists, is unique, and $x_k^* = L/K$ for all k. We then prove that the optimal K^* is as stated in the theorem.

Given any integer $K > 0$, by (9), the KKT condition [15] for the inner optimization problem in (10a) implies that the optimum $x^* = (x_1^*, \ldots, x_K^*)$ satisfies, for all $k = 1, \ldots, K$,

$$h'(x_k^*) = \frac{1}{\overline{F}(x_k^* + \phi)} + (x_k^* + \phi)\frac{f(x_k^* + \phi)}{\left(\overline{F}(x_k^* + \phi)\right)^2} = \mu \quad (13)$$

where μ is a Lagrange multiplier associated with (10b), independent of k. By assumption A1, $\lambda(x) = f(x)/\overline{F}(x)$ is non-decreasing. Moreover $1/\overline{F}(x)$ is non-decreasing, and $x/\overline{F}(x)$ is strictly increasing. Therefore $h'(x)$ is strictly increasing, which is equivalent to $h(x)$ being strictly convex. Thus the inner minimization problem is strictly convex and the KKT condition is also sufficient. A unique solution $x^* = (x_1^*, \ldots, x_K^*)$ exists. Moreover, since all x_k^* are uniquely determined by (13), they are the same and hence $x_k^* = L/K$ for all k.

This reduces the minimization (10) to:

$$\min_K K\frac{L/K + \phi}{\overline{F}(L/K + \phi)} \;=\; \min_K L\frac{L/K + \phi}{L/K\overline{F}(L/K + \phi)}$$

Since L is constant, this is equivalent to solving

$$x^* = \arg\min_x g(x), \qquad x = \left\{L, \frac{L}{2}, \frac{L}{3}, \ldots\right\} \quad (14)$$

where g is defined in (11). The derivative of $g(x)$ is

$$\frac{dg(x)}{dx} = \frac{(x^2 + \phi x)f(x + \phi) - \phi\overline{F}(x + \phi)}{\left(x\overline{F}(x + \phi)\right)^2}$$

Since $\lambda(x) = f(x)/\overline{F}(x)$ is continuous by assumption, and since $\lim_{x\to 0} g(x) = \infty$ and $\lim_{x\to\infty} g(x) = \infty$, an optimal $x^* \in \{L, L/2, L/3, \ldots\}$ and hence optimal K^* exists. Moreover, any unconstrained minimum a of $g(x)$ must also be a stationary point. But $g'(x) = 0$ for x satisfying

$$\xi(x) \;:=\; \frac{f(x + \phi)}{\overline{F}(x + \phi)} \cdot \frac{x(x + \phi)}{\phi} \;=\; 1.$$

Since $f(x + \phi)/\overline{F}(x + \phi)$ is non-decreasing, $x(x + \phi)/\phi$ is strictly increasing, $\xi(0) = 0$, $\lim_{x\to\infty} \xi(x) = \infty$, and $f(x)$ is continuous, it follows that the equation $\xi(x) = 1$ will have a unique solution, which is the unique minimizer a of $g(x)$ defined in (12). Moreover, it implies that $g(x)$ is unimodal. This means that K^* equal to $\lfloor L/a \rfloor$ or $\lceil L/a \rceil$, whichever produces a smaller $g(x)$ value. ∎

Note that since $g(0) = \infty$, the theorem implies that $K^* = 1$ if $L \le a$.

[16] provides a useful sufficient condition for Assumption A1: if f is log-concave, so is F. Since F is log-concave if and only if its failure rate is non-decreasing, a log-concave f satisfies A1. The result of Theorem 4 applies to two failure models described in [10, pp. 131] — a model for satellite communication wherein A_i is exponentially distributed (i.e., bit errors occur according to a Poisson process) and a model for terrestrial communication, wherein A_i has a uniform distribution.

We now show that, when L is large, the unique optimal fragment size x^* is close to a; indeed, x^* approaches a as L increases.

Lemma 10. *Suppose $L > a$. Under Assumption A1, the optimal fragment size $x^*(L)$ satisfies:*

1) $a/2 < x^*(L) \le 2a$.
2) $a/(1 + a/L) < x^*(L) \le a/(1 - a/L)$.

Proof: We know that for some integer K:

$$\frac{L}{K + 1} \le a < \frac{L}{K} \quad (15)$$

and

$$x^* \;=\; \frac{L}{K} \quad \text{or} \quad x^* \;=\; \frac{L}{K + 1}$$

In the first case, $x^*K/(K + 1) \le a < x^*$ implying $x^*/2 \le a < x^*$, i.e., $a < x^* \le 2a$. In the second case, $x^* \le a < x^*(K + 1)/K \le 2x^*$ implying $a/2 < x^* \le a$. Combining these yields $a/2 < x^* \le 2a$.

From (15) we get

$$\frac{L}{a} - 1 \le K < \frac{L}{a}$$

implying

$$a < \frac{L}{K} \le \frac{a}{1 - a/L} \qquad \text{and} \qquad \frac{a}{1 + a/L} < \frac{L}{K + 1} \le a$$

Hence

$$\frac{a}{1 + a/L} < x^* \le \frac{a}{1 - a/L}$$

∎

This admits the following useful corollary.

Corollary 2.

$$\lim_{L\to\infty} x^*(L) = a.$$

B. Simple blind policy $x(l) = \min\{a, l\}$

The optimal fragmentation policy in Theorem 4 depends on the file size L. Consider the L-independent blind policy $x(l) = \min\{a, l\}$ where the fragment size a, given by (12), is always used until the remaining file size drops below a when it is transmitted in a single packet. We will again abuse notation and use a to denote both this blind policy and the fragment size under this policy. Let $J^a(L)$ denote the expected file completion time under policy a when the file size is L. Recall that $J^*(L)$ denotes the minimum expected completion time. From Corollary 2, we know that policy a is asymptotically optimal, i.e., $x^*(L) \to a$. Hence we would expect $J^a(L)$ and $J^*(L)$ to be close for large L. The following result bounds their distance by a constant independent of L, namely the expected time to transmit a packet of size a.

Lemma 11. *Under Assumption A1, for any $L > 0$,*

$$\begin{aligned} 0 \;\le\; J^*(L) - Lg^* \;&\le\; h(a) \\ J^a(L) - J^*(L) \;&\le\; h(a) \end{aligned}$$

where $h(x)$ is defined in (9) and $g^* := g(a)$ is defined by (11) *and* (12).

Proof: If $L = ka$ for some integer k, the proof of Theorem 4 shows that the policy a is optimal, in which case $J^a(L) = J^*(L)$. Suppose then that $ka < L < (k+1)a$ for some integer k. Clearly, $J^a(L) = kh(a) + h(L - ka)$. Since h is monotone, we have

$$kh(a) \le J^a(L) \le (k+1)h(a) \tag{16}$$

Since $J^*(L)$ is monotone in L, we have

$$kh(a) = J^*(ka) \le J^*(L) \le J^*((k+1)a) = (k+1)h(a) \tag{17}$$

Combining (16) and (17), we get that $J^a(L) - J^*(L) \le h(a)$. This proves the sub-optimality bound. Moreover, (17) also implies $Lg^* \le J^*(L) \le Lg^* + h(a)$, as desired. ∎

We make the following remarks:

1) Under both the optimal policy x^* and the blind policy a, the expected completion time grows (roughly) linearly in the file size, the approximating proportionality constant being the minimum per-bit cost $g(a)$.
2) The sub-optimality in expected completion time under the blind policy a is bounded by a constant independent of the file size.

C. Tail asymptotics under policies x^* and a

Denote by $T^*(L)$ and $T^a(L)$ respectively the completion times under the policies x^* and a.

Theorem 5. 1) *If L is light-tailed, then $T^*(L)$ and $T^a(L)$ are light-tailed.*
2) *If $L \in \mathcal{RV}(\alpha)$, then*

$$P\left(T^*(L) > t\right) \sim P\left(T^a(L) > t\right) \sim P\left(L > \frac{t}{g(a)}\right)$$

Since the blind policy a belongs to the class of constant fragmentation policies (see Section III), the tail asymptotics of $T^a(L)$ stated in the theorem follow from Theorem 3. Lemma 10 implies that the optimal policy x^* is a bounded fragmentation policy (see Section III). It follows then from Theorem 2 that $T^*(L)$ is light-tailed if L is light-tailed. However, the exact tail asymptotics of $T^*(L)$ when $L \in \mathcal{RV}(\alpha)$ claimed above requires a separate proof, which we provide in Appendix B.

V. Robustness to failure process

Although the blind policy of Section IV-B does not require knowledge of the file size L, it assumes knowledge of the statistics of the failure process $(A_n, n \ge 1)$. In this section, we derive bounds on the penalty for applying either the optimal policy x^* or blind policy a of Section IV designed for a failure distribution $\hat{F}$, when the actual distribution is F. Variables with a hat will be used to denote quantities defined with respect to $\hat{F}$, e.g., $\hat{a}$ and $\hat{x}^*$ are the the blind and optimal policy, respectively, for the design distribution $\hat{F}$, while a and x^* are those for the true distribution F. Further, let $g^* := g(a) = \min_x g(x)$ where g is defined in (11).

We will compare the expected cost $J^{\hat{a}}(L)$ under F of the blind policy $\hat{a}$ designed for $\hat{F}$, and the expected cost $J^{\hat{x}^*}(L)$ under F of the policy $\hat{x}^*$ optimal for $\hat{F}$, with the true minimum cost $J^*(L)$. The following result specifies the cost increment in terms of the per-bit cost function g defined in (11).

Theorem 6. *Under assumption A1*

$$\begin{aligned}\lim_{L\to\infty} \frac{J^{\hat{a}}(L) - J^*(L)}{L} &= g(\hat{a}) - g^* \\ \lim_{L\to\infty} \frac{J^{\hat{x}^*}(L) - J^*(L)}{L} &= g(\hat{a}) - g^*\end{aligned}$$

Proof: To establish the first limit, note that for any constant fragment size x,

$$J^x(L) = \left\lfloor \frac{L}{x} \right\rfloor xg(x) + x'g(x'),$$

where $x' = (L - \lfloor L/x \rfloor\, x) \in [0, x)$. Since $x'g(x') = h(x')$, and $h(\cdot)$ is non-decreasing, this implies

$$|J^x(L) - Lg(x)| \;<\; h(x). \tag{18}$$

We also have $Lg^* \le J^*(L) \le Lg^* + h(a)$ from Lemma 11. Setting $x = \hat{a}$ in (18) then gives

$$J^{\hat{a}}(L) - J^*(L) \;=\; L\left(g(\hat{a}) - g^*\right) + \alpha(L)h(\max(a, \hat{a}))$$

for some $\alpha : \mathbb{R}_+ \to (-1, 1)$. Dividing the inequality by L and taking the limit as $L \to \infty$ gives the result.

The second inequality follows by setting $x = \hat{x}^*$ in (18) and following the same argument, noting that $\hat{x}^* \to \hat{a}$ and g is continuous. ∎

We make two remarks. First, without modeling error, $\hat{F} = F$, Lemma 11 implies that the per-bit cost penalty approaches zero as L increases. With modeling error, this penalty approaches $g(\hat{a}) - g^*$ which has the intuitive interpretation that the per-bit cost over the entire file approaches the per-bit cost over a packet. Second, an immediate corollary of Theorem 6 is that the overall per-bit costs of policies $\hat{a}$ and $\hat{x}^*$ are asymptotically the same, i.e.

$$\lim_{L\to\infty} \frac{J^{\hat{a}} - J^{\hat{x}^*}}{L} \;=\; 0 \tag{19}$$

which is also intuitive given $\hat{x}^* \to \hat{a}$.

The limit $g(\hat{a}) - g^*$ in Theorem 6 implies a bound on the per-bit cost penalty in terms of the error bound between the design distribution $\hat{F}$ and the true distribution F. Specifically, suppose the tail distributions satisfy

$$1 - F(x) = (1 - \hat{F}(x))(1 + \Delta(x)) \tag{20a}$$

where

$$-\Delta_{\min} \le \Delta(x) \le \Delta_{\max} \tag{20b}$$

for some $\Delta_{\min}$ and $\Delta_{\max}$. In that case, the cost penalty can be quantified in terms of the known quantities $\hat{g}^* := \hat{g}(\hat{a}) = \min_x \hat{g}(x)$, $\Delta_{\min}$ and $\Delta_{\max}$.

Theorem 7. *Under assumption A1*

$$\lim_{L\to\infty} \frac{J^{\hat{a}}(L) - J^*(L)}{L} \le \frac{\Delta_{\max} + \Delta_{\min}}{(1 + \Delta_{\max})(1 - \Delta_{\min})}\, \hat{g}^*$$

Proof: By Theorem 6, it suffices to show that the right hand side is at least $g(\hat{a}) - g^*$. By insertion of equation (20) into equation (11) we see that

$$\frac{\hat{g}(x)}{1 + \Delta_{\max}} \le g(x) \le \frac{\hat{g}(x)}{1 - \Delta_{\min}} \tag{21}$$

Since equation (21) holds for a, we get

$$\frac{\hat{g}^*}{1+\Delta_{\max}} \leq \frac{\hat{g}(a)}{1+\Delta_{\max}} \leq g^*. \tag{22}$$

Since it also holds for $\hat{a}$, we get

$$g(\hat{a}) \leq \frac{\hat{g}^*}{1-\Delta_{\min}}. \tag{23}$$

Combining inequalities (22) and (23), we get

$$\begin{aligned} g(\hat{a}) - g^* &\leq \frac{\hat{g}^*}{1-\Delta_{\min}} - \frac{\hat{g}^*}{1+\Delta_{\max}} \\ &= \frac{\Delta_{\max}+\Delta_{\min}}{(1+\Delta_{\max})(1-\Delta_{\min})}\,\hat{g}^* \end{aligned}$$

as required. ∎

Corollary 3. *If $\Delta_{\min} = \Delta_{\max}$, under assumption A1,*

$$\lim_{L\to\infty} \frac{J^{\hat{a}}(L) - J^*(L)}{L} \leq \frac{2\Delta_{\max}}{1-(\Delta_{\max})^2}\,\hat{g}^*$$

VI. A MODEL FOR CHECKPOINTING ON AN UNRELIABLE SERVER

In this section, we introduce a model for job fragmentation/checkpointing on an unreliable server [7]–[9]. This model is a variant of the file fragmentation model described earlier, and the results we have proved for the file fragmentation model can be extended to this checkpointing model. Since these results make an independent contribution to the checkpointing literature (see Section VII), we state them in this section, and also describe other scenarios where these results are applicable.

A. Model

Consider a server that alternates between states of availability and unavailability according to a semi-Markov process. This can model, for example, a server that is prone to failure: the unavailability period corresponding to the server downtime after a failure. The server availability (unavailability) periods are distributed as A (U) respectively. A job of random size L, independent of the server availability process, is to be processed by the server. If the server becomes unavailable when the job is still being processed, we assume that the job needs to be restarted from the beginning, i.e., the work that is partially completed is lost. This is the RESTART model in queueing literature (see [4]). Recently, the following result has been proved about the job completion time under RESTART [4], [6].

Lemma 12 ([4], [6])**.** *Under RESTART, if the distribution of the job size L has unbounded support, then the job completion time is heavy-tailed. Moreover, if $L \in \mathcal{RV}(\alpha)$ and A is light-tailed, then the completion time $T(L)$ satisfies*

$$\lim_{t\to\infty} \frac{-\log P\left(T(L) > t\right)}{\log(t)} = 0.$$

This means that under RESTART, the completion time tail is asymptotically much heavier than the job size tail. Intuitively, this is because large jobs get restarted many times before they complete, and therefore have disproportionately large completion times.

Lemma 12 obviously motivates the use of fragmentation/checkpointing strategies to reduce the job completion time.[4] Accordingly, let us now consider the RESTART model allowing for job fragmentation. We assume from this point on that the server availability periods are exponential, i.e., A is exponential with mean $1/\mu$. We let arbitrary portions/fragments of the job to be submitted to the server at a time. However, there is a fragmentation cost $\phi > 0$, i.e., the processing time of a submitted fragment gets padded by ϕ. One interpretation of ϕ is the check-pointing overhead, i.e., the time taken to save the current state of the job to disk. Of course, if the server becomes unavailable before the submitted fragment completes processing, then no useful work gets done and we submit another fragment when the server becomes available again. We may model the job submission process as follows. Let $\{t_n\}_{n\geq 1}$ denote the instants at which we make fragment submissions to the server; $t_1 = 0$, t_n is the time instant of the nth submission. Let l_n denote the size of the remaining (yet unprocessed) part of the job at time t_n, with $l_1 = L$. At time t_j, we submit a fragment of work of size x_n to the server. The l_n evolve as follows.

$$l_{n+1} = l_n - x_n \mathbf{1}(x_n + \phi \leq A_n)$$

Note that the fragment size is measured in terms of its processing time. Here, $\{A_n\}_{n\geq 1}$ is an i.i.d. sequence of random variables distributed as A. Note that since the server availability periods are memoryless (exponential), the nth submission completes successfully with probability $P\left(A_n \geq x_n + \phi\right)$, independent of past and subsequent submissions. In the event of a failure, i.e., $A_n < x_n + \phi$, $t_n + A_n$ is interpreted as the instant of server failure.

The cost τ_n accumulated at time-step n is simply the time until the next submission, i.e., $t_{n+1} - t_n$.

$$\begin{aligned} \tau_n = (\ &(x_n+\phi)\mathbf{1}\left(A_n \geq x_n + \phi\right) \\ &+ (A_n + U_n)\mathbf{1}\left(A_n < x_n + \phi\right)\ )\ \ \mathbf{1}(l_n > 0) \end{aligned}$$

Here, $\{U_j\}_{j\geq 1}$ is an IID sequence distributed as U, independent of $\{A_j\}_{j\geq 1}$. This equation is to be interpreted as follows. If the fragment submitted at t_j gets processed to completion, then the time cost is just the processing time $x_j + \phi$. However, if the server becomes unavailable before the fragment is processed, the time cost is sum of the time until the server became unavailable A_j, and an unavailability period U_j. Note that the above cost model differs from that for the file fragmentation model (3) in the event of a failure. Specifically, in the above model, failure is detected immediately, whereas in the file fragmentation model, failure is detected only after the fragment processing time. Moreover, the present model allows for a non-negligible unavailable time after a failure.

[4]Intuitively, a good fragmentation policy would seek to minimize the lost work in each availability period, such that the completion time is close to that under the RESUME model [17], [18]. In the RESUME model, the job is simply 'paused' when the server becomes unavailable, and is resumed from the same point (with no overhead) when the server becomes available again.

Finally, the job completion time $T(L)$ is given by

$$T(L) = \sum_{n \geq 1} \tau_n.$$

This completes the model description. Note that the above modeling assumptions are standard in the checkpointing literature (see, for example, [7]–[9], [19], [20]).

B. Results

The results in Sections III and IV for the file fragmentation model extend naturally to the job checkpointing model. We state the main results here.

As in the file fragmentation model, it can be proved that checkpointing policies that generate independent or bounded fragments guarantee: i) light-tailed completion times for light-tailed job sizes, ii) optimal (in the order sense) completion time tails for regularly varying job sizes. In light of Theorem 12, this means that a large class of checkpointing policies 'lighten' the completion time tail relative to no checkpointing. To keep this presentation brief, we state here only the theorem for the case of independent fragmentation.

Theorem 8 (Independent fragmentation)**.** *Suppose that $\{X_j\}_{j \geq 1}$ is an i.i.d. sequence of strictly positive random variables independent of L and the server availability process. Under the fragmentation policy $x_n = \min\{X_n, l_n\}$,*

1) *If L and U are light-tailed, then $T(L)$ is light-tailed.*
2) *If U is light-tailed and $L \in \mathcal{RV}(\alpha)$, then $P(T(L) > t) \lesssim P\left(L > \frac{t}{\sigma}\right)$ where*

$$\sigma = \frac{\mathbb{E}[A_1 + U_1](1 - P(A_1 \geq X_1 + \phi))}{P(X_1 + \phi \leq A_1)\,\mathbb{E}[X_1 | X_1 + \phi \leq A_1]}.$$

Next, we turn to the problem of minimizing the average completion time. As before, it turns out that the optimal policy creates equally sized fragments; i.e., equally spaced checkpoints. To describe the optimal policy, we need the following definitions:

1) $h(x) = (\mathbb{E}[A_1] + \mathbb{E}[U_1])\left(e^{\mu(x+\phi)} - 1\right)$,
2) $g(x) = h(x)/x$,
3) $a = \arg\min_{x>0} g(x)$.

These definitions are parallel to those for the file fragmentation model. Specifically, $h(x)$ is the expected time for completion of a fragment of size x (assuming it is (re)submitted until it completes). $g(x)$ is therefore the cost per unit fragment size for a fragment of size x, and a is the fragment size that leads to the minimum cost per unit fragment size. In terms of these quantities, the optimal policy is identical to that for the file fragmentation model:

Theorem 9. Optimal fragmentation policy x^*: *The expected job completion time is minimized by fragmenting the job into K^* fragments of equal size $x^* = \frac{L}{K^*}$, where K^* is given by*

$$K^* = \begin{cases} 1 & \text{for } L \leq a \\ \underset{k \in \{\lceil \frac{L}{a} \rceil, \lfloor \frac{L}{a} \rfloor\}}{\arg\min} \; g(L/k) & \text{for } L > a \end{cases}.$$

Each fragment is (re)submitted to the server till it gets processed completely. Denote the completion time under the optimal policy by $T^*(L)$. Then

1) *If L and U are light-tailed, then $T^*(L)$ is light-tailed.*
2) *If U is light-tailed, and $L \in \mathcal{RV}(\alpha)$, then $P(T^*(L) > t) \sim P\left(L > \frac{t}{g(a)}\right)$.*

It is important to note that the optimal fragmentation policy does not depend on the server unavailability period distribution U.

Finally, as before, it is possible to fragment close to optimally while remaining blind to the job size.

Theorem 10. Sub-optimal blind fragmentation policy a: *Consider the following simple fragmentation policy u_a.*

$$x_j = \min\{l_j, a\}$$

For a job of size l, let $J^a(l)$ denote the expected completion time under this policy. Then

$$J^a(l) - J^*(l) \leq h(a),$$

where $J^*(l)$ is the expected completion time under the optimal policy for a job of size l. Let $T^a(L)$ denote the completion time under this policy. Then

1) *If L and U are light-tailed, then $T^a(L)$ is light-tailed.*
2) *If U is light-tailed, $L \in \mathcal{RV}(\alpha)$, then $P(T^a(L) > t) \sim P\left(L > \frac{t}{g(a)}\right)$.*

C. Scenarios of Applicability

In addition to the case of an unreliable server in a computing environment, the model and results of this section are also applicable to the following scenarios.

1) Priority queue: Consider a queue that serves jobs of two priority levels. Low priority jobs use the server when there are no high priority jobs in the system. If a high priority job arrives when the server is processing an low priority job, the low priority job is pre-empted and needs to be restarted. In this scenario, our job fragmentation model applies to a low priority job. U denotes the busy period induced by high priority jobs and A denotes the time between these busy periods. If high priority jobs arrive as per a Poisson process, then A is exponential.
2) File fragmentation in cognitive radio setting: Consider a secondary user that is allowed to use a wireless channel to transfer its file of size L whenever primary (high priority) users are not using it. The secondary user must abort its transmission whenever primary users want to use the channel. Our model corresponds to file fragmentation for the secondary user. The availability period for the secondary users will be exponentially distributed if the primary users initiate transfers according to a Poisson process.

VII. Related work

The work in this paper is motivated by recent work [3]–[6] which showed that heavy-tailed completion times can result from RESTART/retransmission mechanisms. Indeed, this effect has subsequently been shown to be robust to several schemes aimed at alleviating it. The fragmentation scheme of [21], which uses the sizes of the previous $k + m$ server

availability periods, lightens the completion time tail by adding k additional moments, but the resulting tail is still heavy. Multipath is explored in [22] to mitigate power-law completion time. It is shown there that redundant routing, where the entire file is sent along multiple paths and the completion time is the time when the first copy arrives at the destination correctly, preserves the power law. Split routing, where disjoint fragments of the file are sent along multiple paths and the completion time is the time when the last fragment arrives, also retains a power-law completion time though the tail can be lightened with a larger index. Having a bounded file size distribution of course eliminates the heavy-tailed completion time; however, it is shown in [23], [24] that when the upper bound on the file size distribution is large, the completion time distribution retains a power-law body. To the best of our knowledge, this work is the first to show that heavy-tailed completion times are actually quite fragile, and can be removed by a large class of simple fragmentation schemes.

In the context of file fragmentation or packet sizing, optimal fragmentation that minimizes average completion time or maximizes throughput is a classical problem. A good reference is the early work [10, pp. 131–134]. However, to the best of our knowledge, completion time tail behavior has not been analysed in this setting (except in the recent work listed above). Of course, studying the completion time tail is particularly relevant in light of the recent results in [3]–[6].

Similarly, there is a sizeable literature on checkpointing; see [19] for an early survey and also references in, e.g., [25]. Considering various model variations, several papers analyse the problem of optimal checkpointing to minimize average completion time; for example, [7]–[9], [19], [20], [25]–[27]. However, to the best of our knowledge, except for the recent work listed above, none of these papers analyse the completion time tail. Once again, we note that an analysis of the completion time tail is particularly relevant in light of the recent results in [3]–[6].

From a practical point of view, tail performance is also of increasing importance. In highly parallel systems, performance in the typical case depends on tail performance, since the overall delay is determined by the maximum delay of many tasks. Consequently, companies like Google make many design decisions based on tail performance [28].

Appendix A
Proof of Lemma 2

Proof of Lemma 2:

Proofs are Statements (1) and (2) are elementary and are omitted.

a) Proof of Statement (3): Let $Z = \sum_{i=1}^{N} X_i$. Pick $\beta \in (0, 1/\mathbb{E}[X_1])$.

$$\begin{aligned} P(Z>t) &= P(Z>t;\ N\le \beta t) + P(Z>t; N>\beta t) \\ &\le P\left(\sum_{i=1}^{\lfloor \beta t\rfloor} X_i > t\right) + P(N>\beta t). \\ &\le P\left(\sum_{i=1}^{\lfloor \beta t\rfloor} X_i > \frac{\lfloor \beta t\rfloor}{\beta}\right) + P(N>\beta t) \\ &=: I + II. \end{aligned}$$

Using the Chernoff bound, we conclude that there exists $\alpha_1 > 0$ such that $I \le e^{-\alpha_1 \lfloor \beta t\rfloor}$. Also, since N is light-tailed, there exists $\alpha_2 > 0$ such that $II \le e^{-\alpha_2 \beta t}$ for large enough t. Since we have an exponentially decaying upper bound on the tail of Z, it follows that Z is light-tailed. This completes the proof of Statement (3).

b) Proof of Statement (4): Define

$$N := \inf\{n \in \mathbb{N} | \sum_{i=1}^{n} X_i \ge L\}.$$

Let us first consider the case that X_i are light-tailed. Pick $\beta \in (0, \mathbb{E}[X_1])$.

$$\begin{aligned} P(N>n) &= P\left(\sum_{i=1}^{n} X_i < L\right) \\ &\le P\left(\sum_{i=1}^{n} X_i < L;\ L > \beta n\right) \\ &\quad + P\left(\sum_{i=1}^{n} X_i < L;\ L \le \beta n\right) \\ &\le P(L>\beta n) + P\left(\sum_{i=1}^{n} X_i < \beta n\right) \\ &=: I + II. \end{aligned}$$

Since L is light-tailed, there exists $\alpha_1 >$ such that $I \le e^{-\alpha_1 n}$ for large enough n. Also, using the Chernoff bound, we conclude that there exists $\alpha_2 > 0$ such that $II \le e^{-\alpha_2 n}$. Since we now have an exponentially decaying upper bound on the tail d.f. of N, it follows that N is light-tailed. This completes the proof in the case that X_i is light tailed.

If instead the X_i are heavy-tailed, then we may define $Y_i = X_i \mathbf{1}(X_i \le y)$ for some $y > 0$ such that $P(Y_i > 0) > 0$. Then $\tilde{N} := \inf\{n \in \mathbb{N} | \sum_{i=1}^{n} Y_i \ge L\}$ is light-tailed. However, since $N \le_{\text{a.s.}} \tilde{N}$, this implies N is light-tailed (from Statement (1) of this lemma). ∎

Appendix B
Proof of Theorem 5: Tail asymptotics of $T^*(L)$

This section is devoted to proving the tail behavior of $T^*(L)$ claimed in the statement of Theorem 5, i.e., we prove that if $L \in RV(\alpha)$ then

$$P(T^*(L) > t) \sim P\left(L > \frac{t}{g(a)}\right). \tag{24}$$

The proof of (24) is based on stochastically bounding the optimal completion time $T^*(l)$ from both sides. We need the following notation. Let $W^{(z)}$ denote a random variable distributed as the time to successfully transmit a fragment of size $z > 0$. Note that $h(z) = \mathbb{E}\left[W^{(z)}\right]$, and that $W^{(z)}$ is stochastically increasing in z. Since $W^{(z)} \stackrel{d}{=} \sum_{i=1}^{N}(z+\phi)$, where N is a geometric random variable with mean $1/P(A_1 \ge z+\phi)$, we infer from Lemma 2 that $W^{(z)}$ is light-tailed. Let $(W_i^{(z)}, i \ge 1)$ denote a sequence of i.i.d. random variables independent of L distributed as $W^{(z)}$. Note that

$$T^*(l) \stackrel{d}{=} \sum_{i=1}^{K^*(l)} W_i^{(x^*(l))}.$$

Now, pick $\epsilon \in (0,a)$. Since $x^*(l) \overset{l\uparrow\infty}{\to} a$ by Corollary 2, there exists an $l_0 > 0$ such that $x^*(l) \in (a-\epsilon, a+\epsilon)$ for all $l \geq l_0$. Note that for $l \geq l_0$,

$$\left\lfloor \frac{l}{a-\epsilon} \right\rfloor \geq K^*(l) \geq \left\lceil \frac{l}{a+\epsilon} \right\rceil.$$

Define

$$\hat{T}(l) := \begin{cases} 0 & \text{for } 0 \leq l < l_0 \\ \sum_{i=1}^{\lceil \frac{l}{a+\epsilon} \rceil} W_i^{(a-\epsilon)} & \text{for } l \geq l_0 \end{cases},$$

$$\tilde{T}(l) := \begin{cases} T^*(l) & \text{for } 0 \leq l < l_0 \\ \sum_{i=1}^{\lfloor \frac{l}{a-\epsilon} \rfloor} W_i^{(a+\epsilon)} & \text{for } l \geq l_0 \end{cases}.$$

It is easy to check that $\hat{T}(l) \leq_{\rm st} T^*(l) \leq_{\rm st} \tilde{T}(l)$ for all l, which implies that

$$\hat{T}(L) \leq_{\rm st} T^*(L) \leq_{\rm st} \tilde{T}(L). \tag{25}$$

The following lemmas characterize the tail asymptotics of $\hat{T}(L)$ and $\tilde{T}(l)$.

Lemma 13. *For the chosen $\epsilon \in (0,a)$,*

$$P\left(\hat{T}(L) > t\right) \sim P\left(L > \frac{t(a+\epsilon)}{h(a-\epsilon)}\right).$$

Lemma 14. *For the chosen $\epsilon \in (0,a)$,*

$$P\left(\tilde{T}(L) > t\right) \sim P\left(L > \frac{t(a-\epsilon)}{h(a+\epsilon)}\right).$$

Now (24) follows by combining (25) and Lemmas 13 and 14. Indeed, it follows from (25) and Lemma 14 that

$$\begin{aligned} \limsup_{t\to\infty} \frac{P\left(T^*(L) > t\right)}{P\left(L > \frac{t}{g(a)}\right)} &\leq \limsup_{t\to\infty} \frac{P\left(\tilde{T}(L) > t\right)}{P\left(L > \frac{t}{g(a)}\right)} \\ &= \limsup_{t\to\infty} \frac{P\left(L > \frac{t(a-\epsilon)}{h(a+\epsilon)}\right)}{P\left(L > \frac{t}{g(a)}\right)} \\ &= \left(\frac{a}{a-\epsilon}\frac{h(a+\epsilon)}{h(a)}\right)^{\alpha}. \end{aligned}$$

The last step above follows from the definition of regular variation. Similarly, it follows from (25) and Lemma 13 that

$$\liminf_{t\to\infty} \frac{P\left(T^*(L) > t\right)}{P\left(L > \frac{t}{g(a)}\right)} \geq \left(\frac{a}{a+\epsilon}\frac{h(a-\epsilon)}{h(a)}\right)^{\alpha}.$$

Now, (24) follows by letting $\epsilon \downarrow 0$.

It remains now to prove Lemmas 13 and 14. We give the proof of Lemma 14 below. The proof of Lemma 13 follows along similar lines, and is omitted.

Proof of Lemma 14: Define $\tilde{L} = L\,\mathbf{1}(L \geq l_0)$. Also, let us take $\tilde{T}(0) = 0$. For $t > 0$,

$$\begin{aligned} P\left(\tilde{T}(L) > t\right) &= P\left(\tilde{T}(L) > t;\ L \geq l_0\right) \\ &\quad + P\left(\tilde{T}(L) > t;\ L < l_0\right) \\ &= P\left(\tilde{T}(\tilde{L}) > t\right) + P\left(\tilde{T}(L) > t;\ L < l_0\right) \\ &=: I + II. \end{aligned}$$

We now study the terms I and II separately. Specifically, we will show that Term I accounts for the claimed tail asymptotics of $\tilde{T}(L)$, while Term II makes an asymptotically negligible contribution.

We start by analyzing Term I, which is defined as the tail of a random sum:

$$I = P\left(\sum_{i=1}^{\lfloor \frac{\tilde{L}}{a-\epsilon} \rfloor} W_i^{(a+\epsilon)} > t\right).$$

Since $\left\lfloor \frac{\tilde{L}}{a-\epsilon} \right\rfloor$ is regularly varying, and $W_i^{(a+\epsilon)}$ is light-tailed, it follows from standard results on tails of random sums (see Theorem A3.20 in [29]) that

$$\begin{aligned} I &\sim P\left(\left\lfloor \frac{\tilde{L}}{a-\epsilon} \right\rfloor > \frac{t}{h(a+\epsilon)}\right) \\ &\sim P\left(\tilde{L} > \frac{t(a-\epsilon)}{h(a+\epsilon)}\right) \\ &\sim P\left(L > \frac{t(a-\epsilon)}{h(a+\epsilon)}\right). \end{aligned}$$

It now remains to prove that $II = o(I)$ as $t \to \infty$. To prove this, it suffices to show that Term II decays exponentially with respect to t. It follows from Theorem 4 and Lemma 10 that for $l \in (0, l_0)$, $K^*(l) \leq \lceil 2l_0/a \rceil$, and $x^*(l) \leq 2a$. This means that for $l \in (0, l_0)$,

$$T^*(l) \leq_{\rm st} Z := \sum_{i=1}^{\lceil 2l_0/a \rceil} W_i^{(2a)}.$$

Now,

$$\begin{aligned} II = P\left(\tilde{T}(L) > t;\ L < l_0\right) &= P\left(T^*(L) > t;\ L < l_0\right) \\ &\leq P\left(Z > t\right). \end{aligned}$$

Since Z is light-tailed, it follows that there exists $\phi > 0$ such that $II \leq e^{-\phi t}$ for large enough t, which implies that $II = o(I)$. ∎

ACKNOWLEDGMENT

We thank Adam Wierman, Lijun Chen and Mani Chandy for helpful discussions. We acknowledge support of ARO through MURI Grant W911NF-08-1-0233, NSF through the NetSE grant, the Caltech Lee Center for Advanced Networking, and Australian Research Council grant DP0985322. The first author also acknowledges support from an NWO VIDI grant.

REFERENCES

[1] J. Nair, M. Andreasson, L. Andrew, S. Low, and J. Doyle, "File fragmentation over an unreliable channel," in *Proceedings of IEEE INFOCOM*, 2010.

[2] J. Nair and S. H. Low, "Optimal job fragmentation," *SIGMETRICS Performance Evaluation Review*, vol. 37, no. 2, pp. 21–23, 2009.

[3] R. Sheahan, L. Lipsky, P. M. Fiorini, and S. Asmussen, "On the completion time distribution for tasks that must restart from the beginning if a failure occurs," *ACM SIGMETRICS Performance Evaluation Review*, vol. 34, no. 3, pp. 24–26, 2006.

[4] S. Asmussen, P. Fiorini, L. Lipsky, T. Rolski, and R. Sheahan, "Asymptotic behavior of total times for jobs that must start over if a failure occurs," *Mathematics of Operations Research*, vol. 33, no. 4, pp. 932–944, 2008.

[5] P. R. Jelenković and J. Tan, "Characterizing heavy-tailed distributions induced by retransmissions," *Advances in Applied Probability*, vol. 45, no. 1, pp. 106–138, 2013.

[6] ——, "Can retransmissions of superexponential documents cause subexponential delays?" in *Proceedings of IEEE INFOCOM*, 2007.

[7] A. Duda, "The effects of checkpointing on program execution time," *Information Processing Letters*, vol. 16, no. 5, pp. 221–229, 1983.

[8] V. Grassi, L. Donatiello, and S. Tucci, "On the optimal checkpointing of critical tasks and transaction-oriented systems," *Software Engineering, IEEE Transactions on*, vol. 18, no. 1, pp. 72–77, Jan 1992.

[9] V. Kulkarni, V. Nicola, and K. S. Trivedi, "Effects of checkpointing and queueing on program performance," *Communications in Statistics – Stochastic Models*, vol. 6, no. 4, pp. 615–648, 1990.

[10] M. Schwartz, *Telecommunication networks: Protocols, modeling and analysis*. Addison-Wesley Longman Publishing Co., 1986.

[11] N. Bingham, C. Goldie, and J. Teugels, *Regular variation*. Cambridge University Press, 1989.

[12] S. Resnick, *Heavy-Tail Phenomena: Probabilistic and Statistical Modeling*. Springer, 2007.

[13] K. Sigman, "Appendix: A primer on heavy-tailed distributions," *Queueing Systems*, vol. 33, no. 1, pp. 261–275, 1999.

[14] F. Guillemin, P. Robert, and B. Zwart, "Tail asymptotics for processor-sharing queues," *Advances in Applied Probability*, vol. 36, no. 2, pp. 525–543, 2004.

[15] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[16] M. Bagnoli and T. Bergstrom, "Log-concave probability and its applications," Department of Economics, University of California Santa Barbara, Economics Working Paper Series, 2004. [Online]. Available: http://ideas.repec.org/p/cdl/ucsbec/1989d.html

[17] V. Kulkarni, V. Nicola, and K. Trivedi, "On modelling the performance and reliability of multimode computer systems," *Journal of Systems and Software*, vol. 6, no. 1, pp. 175–182, 1986.

[18] ——, "The completion time of a job on multimode systems," *Advances in Applied Probability*, pp. 932–954, 1987.

[19] K. Chandy, "A survey of analytic models for rollback and recovery strategies," *Computer*, vol. 8, no. 5, pp. 40–47, 1975.

[20] E. Coffman Jr and E. Gilbert, "Optimal strategies for scheduling checkpoints and preventive maintenance," *IEEE Transactions on Reliability*, vol. 39, no. 1, pp. 9–18, 1990.

[21] P. R. Jelenković and J. Tan, "Dynamic packet fragmentation for wireless channels with failures," in *Proceedings of ACM MobiHoc*, 2008.

[22] J. Tan, W. Wei, B. Jiang, N. Shroff, and D. Towsley, "Can multipath mitigate power law delays? - Effects of parallelism on tail performance," *SIGMETRICS Performance Evaluation Review*, vol. 38, no. 1, pp. 381–382, 2010.

[23] J. Tan and N. B. Shroff, "Transition from heavy to light tails in retransmission durations," in *Proceedings of IEEE INFOCOM*, 2010.

[24] P. R. Jelenković and E. D. Skiani, "Distribution of the number of retransmissions of bounded documents," *arXiv preprint arXiv:1210.8421*, 2012.

[25] Y. Ling, J. Mi, and X. Lin, "A variational calculus approach to optimal checkpoint placement," *IEEE Transactions on Computers*, vol. 50, no. 7, pp. 699–708, 2001.

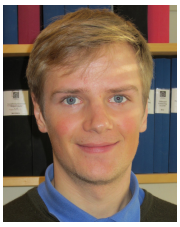
[26] P. Lécuyer and J. Malenfant, "Computing optimal checkpointing strategies for rollback and recovery systems," *IEEE Transactions on Computers*, vol. 37, no. 4, pp. 491–496, 1988.

[27] A. Ziv and J. Bruck, "An on-line algorithm for checkpoint placement," *IEEE Transactions on Computers*, vol. 46, no. 9, pp. 976–985, 1997.

[28] J. Dean and L. A. Barroso, "The tail at scale," *Communications of the ACM*, vol. 56, no. 2, pp. 74–80, 2013.

[29] P. Embrechts, T. Mikosch, and C. Klüppelberg, *Modelling extremal events: for insurance and finance*. London, UK: Springer-Verlag, 1997.

Jayakrishnan Nair received his BTech and MTech in Electrical Engg. (EE) from IIT Bombay (2007) and Ph.D. in EE from California Inst. of Tech. (2012). He has held post-doctoral positions at California Inst. of Tech. and Centrum Wiskunde & Informatica. He is currently an Assistant Professor in EE at IIT Bombay. His research focuses on modeling, performance evaluation, and design issues in queueing systems and communication networks.



Martin Andreasson received the M.Sc. degree in engineering physics KTH Royal Institute of Technology, Stockholm, Sweden, in 2011. He is currently a PhD Student at the Automatic Control Laboratory, KTH Royal Institute of Technology, Stockholm, Sweden. His research interests include distributed control of multi-agent systems, and control of power systems.

Lachlan Andrew (M97-SM05) received the B.Sc., B.E. and Ph.D. degrees in 1992, 1993, and 1997, from the University of Melbourne, Australia. Since 2008, he has been an associate professor at Swinburne University of Technology, Australia, and since 2010 he has been an ARC Future Fellow. From 2005 to 2008, he was a senior research engineer in the Department of Computer Science at Caltech. Prior to that, he was a senior research fellow at the University of Melbourne and a lecturer at RMIT, Australia. His research interests include energy-efficient networking and performance analysis of resource allocation algorithms. He was co-recipient of the best paper award at IGCC2012, IEEE INFOCOM 2011 and IEEE MASS 2007. He is a member of the ACM.

Steven H. Low (F'08) is a Professor of the Department of Computing & Mathematical Sciences and the Department of Electrical Engineering at Caltech. Before that, he was with AT&T Bell Laboratories, Murray Hill, NJ, and the University of Melbourne, Australia. He was a co-recipient of IEEE best paper awards, the R&D 100 Award, and an Okawa Foundation Research Grant. He is a Senior Editor of the IEEE Transactions on Control of Network Systems and the IEEE Transactions on Network Science & Engineering, is on the editorial boards of NOW Foundations and Trends in Networking, and in Electric Energy Systems, as well as Journal on Sustainable Energy, Grids and Networks. He received his B.S. from Cornell and PhD from Berkeley, both in EE.

Professor of Control and Dynamical Systems, Electrical Engineer, and BioEngineering at Caltech. BS, MS EE, MIT (1977), PhD, Math, UC Berkeley (1984). Research is on mathematical foundations for complex networks with applications in biology, technology, medicine, ecology, and neuroscience. Paper prizes include IEEE Baker, IEEE Automatic Control Transactions (twice), ACM Sigcomm, and ACC American Control Conference. Individual awards include AACC Eckman, IEEE Control Systems Field, and IEEE Centennial Outstanding Young Engineer Awards. Has held national and world records and championships in various sports.